



July 2019
IPT Training XML Technologies

Extensible Markup Language (XML)

Trayan Iliev
tiliev@iproduct.org
<http://iproduct.org>

Copyright © 2003-2019 IPT - Intellectual
Products & Technologies

За лектора



Trayan Iliev

- CEO of IPT – Intellectual Products & Technologies (<http://ipproduct.org/>)
- Oracle® certified programmer 15+ Y
- end-to-end reactive fullstack apps with Java, ES6/7, TypeScript, Angular, React and Vue.js
- SOA & BPM with XML, WSDL, WS-*, BPEL, BPMN, UML
- 12+ years IT trainer
- Voxxed Days, jPrime, jProfessionals, BGOUG, BGJUG, DEV.BG speaker
- Organizer RoboLearn hackathons and IoT enthusiast (<http://robolearn.org>)

Какво ще усвоите по време на курса? (1)

- ❖ *Extensible Markup Language (XML)* – характеристики, предназначение, добре конструиран XML, заглавна част, елементи и атрибути, дърво на документа, типове възли (1 ч.)
- ❖ Пространства от имена и тяхното използване (*XML Namespaces*) – префикси и пълно-квалифицирани имена на елементи и атрибути, коментари, инструкции за обработка, *XML Entities*, *CDATA* секции, визуализация в уеб-браузър, CSS – използване с HTML и XML. Приложения на XML. (2 ч.)

Какво ще усвоите по време на курса? (2)

- ❖ Запознаване с езика *JavaScript (ECMAScript)*. Интеграция с HTML, `<script>` таг. Разработка на JS проекти с *Node.js* и *VS Code*. Разработка, стартиране и отстраняване на грешки в JS програми. Стойности, оператори, изрази, ключови думи и коментари. Променливи. Типове данни: *String, Number, Boolean, Array, Object, Function, Math, Date*, и др. Изрази и оператори. Основни езикови конструкции: *if-else, return, switch-case-default, dowhile, for, for-in, with, break, continue, throw, try-catch-finally*. Функции и обекти. Работа с числа, низове и дати. Работа с масиви – основни методи на класа *Array*, сортиране, нови методи. Регулярни изрази. Обектно-ориентиран JS – примитивни типове и обекти, достъп по референция, свойства функции и методи, използване на *this*, методи *call, apply* и *bind*. Наследяване, полиморфизъм и предефиниране. Новости в ECMAScript 6 – 9. (5 ч.)

Какво ще усвоите по време на курса? (3)

- ❖ Основи на HTTP протокола – методи, заглавни части, статус кодове, медийни типове. Асинхронни заявки за достъп до ресурси (*AJAX*) и динамично обновяване на страницата чрез *XMLHttpRequest*. *Fetch API*. XML базирани уеб услуги в стил *SOAP*, *WSDL* и *REST*. *JSON* сериализация на данни. (2 ч.)
- ❖ Осъществяване на достъп и обработка на XML документи от приложни програми. Обектен модел на документа – *Document Object Model (DOM)*. Програмна обработка на XML документи с *DOM API* и *JavaScript* в уеб браузър. (2 ч.)
- ❖ Валидиране на XML документи – *Document Type Definition (DTD)*. Усъвършенствана валидация с *XML Schema Definition (XSD)*. Основни принципи. Типове данни. Структури. Асоцииране на схеми с XML документи. Примери за използване и валидация на XML – *SOAP*, *WSDL*. (4 ч.)

Какво ще усвоите по време на курса? (4)

- ❖ Извличане на секции от XML документи с XPath. Типове XPath възли (nodes). Изрази за селектиране на XPath възли. XPath предикати и wildcards [`*`, `@*`, `node()`]. Селектиране на множество пътища. XPath оси (axes) на селекция. Абсолютни и релативни пътища. XPath оператори и стандартни функции. Практически примери и задачи с XPath. (5 ч.)
- ❖ Extensible Stylesheet Language (XSL). Extensible Stylesheet for Transformation (XSLT). Практическо използване на XPath селектиращи изрази с XSLT. Генериране на HTML от XML. Примери. (2 ч.)
- ❖ Свързване на XML документи – стандарти XLink и XPointer. XQuery – за осъществяване на заявки (подобни на SQL) към XML данни. (3 ч.)
- ❖ Финален тест. Въпроси и отговори. (1 ч.)

График на тренинга

Понеделник (22.07) – четвъртък (25.07):

- ❖ Блок 1: 9.00 - 11.00
- ❖ Почивка: 11.00 - 11.15
- ❖ Блок 2: 11.15 - 13.15
- ❖ Обяд: 13.15 - 14.15
- ❖ Блок 3: 14.15 - 15.45
- ❖ Почивка: 15.45 - 16.00
- ❖ Блок 4: 16.00 - 17.30

Материали и програмни примери

XML Technologies – примери и проекти в
GitHub:

<https://github.com/iproduct/course-xml>

Extensible Markup Language (XML)

- ❖ Официален стандарт – XML 1.0 и XML 1.1 на World Wide Web Consortium - W3C
- ❖ Двоични и текстови файлове, метаданни
- ❖ Standard Generalized Markup Language – SGML
- ❖ HyperText Markup Language – HTML
- ❖ Extensible Markup Language – XML
- ❖ Extensible HyperText Markup Language – XHTML
- ❖ XML – стандарт за създаване на маркиращи езици за конкретни области на приложение
- ❖ Какво значи разширяем - “Extensible”?
- ❖ XML парсъри

Сравнение между HTML и XML

HTML:

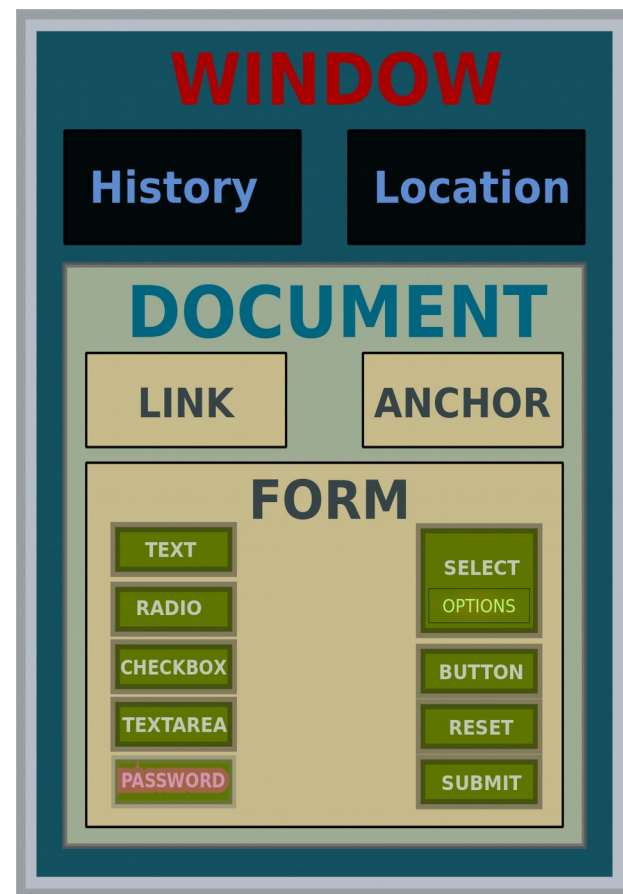
```
<!DOCTYPE html>
<html>
  <head>
    <title>Page of John Smith
  </title>
</head>
<body>
  <h1>John Smith</h1>
</body>
</html>
```

XML:

```
<?xml version="1.0"
encoding="UTF-8"?>
<name>
  <first>John</first>
  <last>Smith</last>
</name>
```

Информационни йерархии – обектни модели

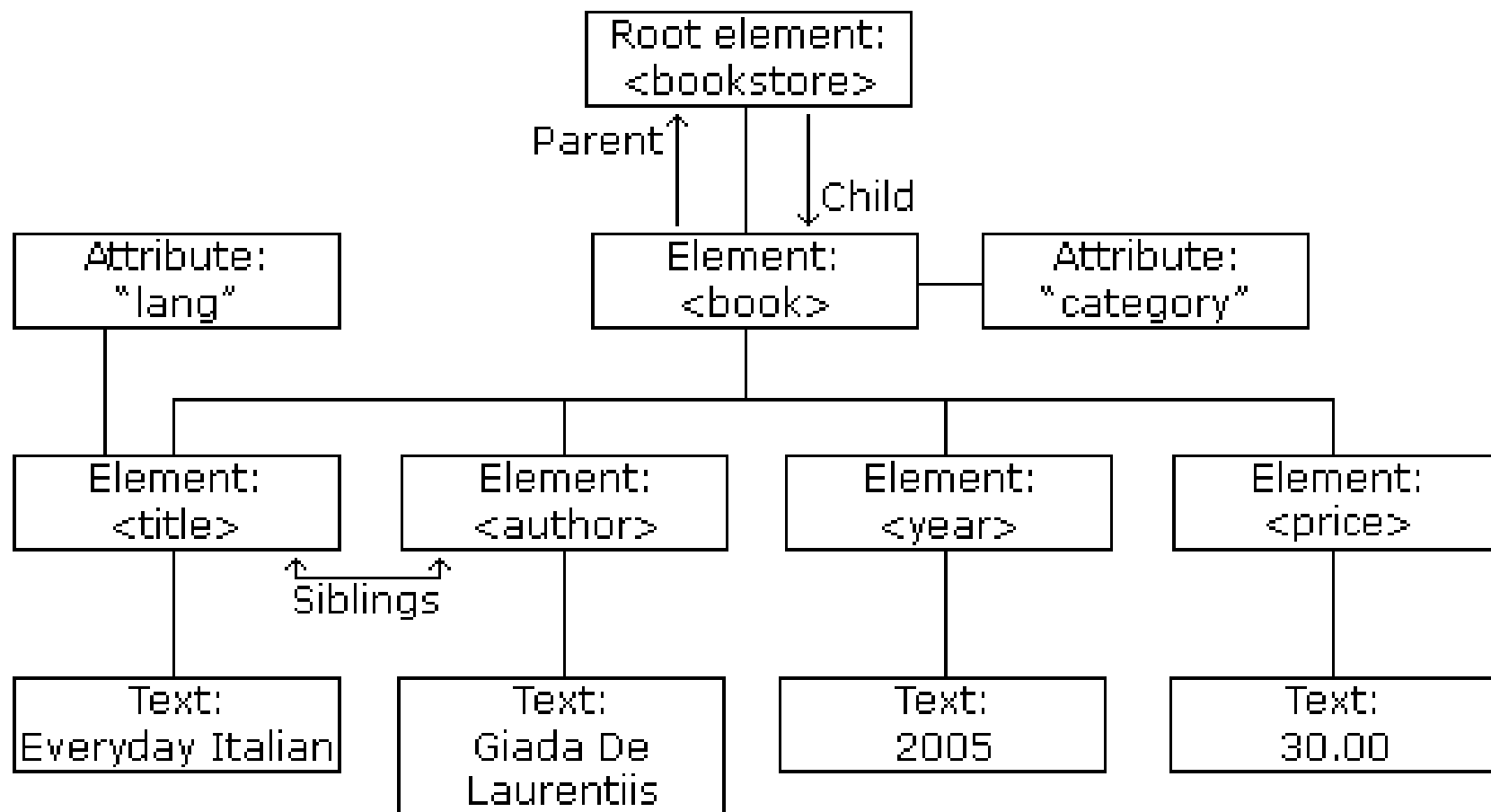
- ❖ Сравнение между HTML и XML – различно предназначение:
 - HTML – специфично приложение: форматиране и визуализиране на информацията
 - XML – няма специфично приложение – произволни информационни структури
- ❖ HTML Document Object Model - DOM
- ❖ XML Document Object Model – DOM



XML документ

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<bookstore>
  <book category="cooking" lang="us-en">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
</bookstore>
```

DOM модел (дърво на документа)



Типове възли (DOM Nodes)

- ❖ **Document** – целия XML документ(the root-node of the DOM tree)
- ❖ **DocumentFragment** – "олекотен" Document обект, част от документ
- ❖ **DocumentType** – тип на документа
- ❖ **ProcessingInstruction** - инструкция за обработка
- ❖ **EntityReference** – референция към Entity
- ❖ **Element** – XML елемент
- ❖ **Attr** – атрибут на елемент
- ❖ **Text** – текстови възел
- ❖ **CDATASection** – CDATA секция (NOT to be parsed)
- ❖ **Comment** – коментар <!-- ... -->
- ❖ **Entity** – представлява ентити
- ❖ **Notation** – представлява нотация

Основни понятия в XML

- ❖ Тагове, елементи и атрибути
- ❖ Дърво на документа – корен, клони, листа, типове възли
- ❖ Елементно съдържание – просто, смесено
- ❖ Документен тип – речник. XML валидация:
 - Document Type Definition (DTD)
 - XML Schema Definition (XSD)
- ❖ Визуализация на XML в уеб браузър:
 - Cascading style Sheets (CSS)
 - Extensible Stylesheet for Transformation (XSLT)
- ❖ XML Namespaces
- ❖ CDATA секции - пр: `<![CDATA[Avg < 90% Response Time(ms)]]>`

Маркър и съдържание, тагове, елементи

- ❖ XML маркър и съдържание – маркър частта се намира между символите **<** и **>** (тагове) или между **&** и **;** (entities), всичко останало е **съдържание**
- ❖ XML процесор (XML parser) и потребителско приложение – анализира и обработва маркърпа и изпраща структурирана информация на потребителското приложение
- ❖ Тагове: **<mytag>**, **</mytag>**, **<mytag />**
- ❖ XML елемент – логически елемент на дървото на документа започващ с отварящ таг и завършващ със затварящ, може да включва съдържание и/или вложени под-елементи:

<mytag>I am**<name>**George**</name>****</mytag>**

Атрибути, XML декларация, валидация

- ❖ Атрибут – представлява двойка име/стойност, която може да се включи в отварящ или празен таг:

```
<book id='15' isbn="817525766-0">
```

Научи Java за един ден

```
</book>
```

- ❖ XML декларация:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

- ❖ **encoding** атрибутът може да има различни стойности: Windows-1252, ISO-8859-1, UTF-8, UTF-16 и др.
- ❖ Добре структуриран XML – един коренен елемент и др.
- ❖ Може да бъде валидиран с DTD, XML Schema, RELAX NG...

Технологии свързани с XML (1)

- ❖ XHTML
- ❖ XML Document Object Model (DOM)
- ❖ Extensible Style Sheet Language (XSL)
- ❖ XSL Transformations (XSLT)
- ❖ XSL Formatting Objects (XSL-FO)
- ❖ XPath – за селектиране на елементи и атрибути в дървото на документа
- ❖ Document Type Definition (DTD)
- ❖ XML Schema Definition (XSD)

Технологии свързани с XML (2)

- ❖ XML Linking Language (**XLink**) и XML Pointer Language (**XPointer**) – за свързване на XML документи
- ❖ **XQuery** – за осъществяване на заявки (подобни на SQL) към XML данни, включително бази от данни
- ❖ **XForms** – HTML форми за въвеждане на данни, независими от конкретната платформа и устройство, част от стандарта XHTML 2.0

Приложения на XML

- ❖ Нов начин за създаване на модулни веб сайтове с възможност за еволюция
- ❖ Семантична мрежа – **RDF**, **OWL**
- ❖ Веб 2.0 – социални технологии, блог, **RSS**, mashup
- ❖ Многослойни веб приложения
- ❖ Веб услуги – **SOAP**, **REST**, **AJAX**
- ❖ Електронен бизнес и търговия – **B2B**, **B2C**

Езици базирани на XML

- ❖ Mathematical Markup Language (MathML)
- ❖ Resource Description Framework (RDF)
- ❖ Web Ontology Language (OWL)
- ❖ Really Simple Syndication (RSS)
- ❖ Synchronized Multimedia Integration Language (SMIL)
- ❖ Scalable Vector Graphics (SVG)
- ❖ Wireless Application Protocol (WAP)
- ❖ Simple Object Access Protocol (SOAP)
- ❖ Web Services Description Language (WSDL)

Пример: MathML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1 plus MathML 2.0//EN"
"http://www.w3.org/Math/DTD/mathml2/xhtml-math11-f.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Example of MathML embedded in an XHTML file</title>
  </head>
  <body>
    <h1>Пример за използване на MathML в XHTML документ</h1>
    <p>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <mi>&#x03C0;</mi> <!-- π -->
        <mo>&#x2062;</mo> <!-- Умножение -->
        <msup>
          <mi>r</mi>
          <mn>2</mn>
        </msup>
      </math> .
    </p>
  </body>
```

Пример: SOAP (1)

❖ SOAP Request:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
```

```
  <S:Header/>
```

```
  <S:Body>
```

```
    <ns2:add xmlns:ns2="http://calculator.me.org/">
```

```
      <i>5</i>
```

```
      <j>12</j>
```

```
    </ns2:add>
```

```
  </S:Body>
```

```
</S:Envelope>
```


Пример: SOAP (2)

❖ SOAP Response:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
```

```
  <S:Body>
```

```
    <ns2:addResponse xmlns:ns2="http://calculator.me.org/">
```

```
      <return>17</return>
```

```
    </ns2:addResponse>
```

```
  </S:Body>
```

```
</S:Envelope>
```

Програмни интерфейси за обработка на XML

- ❖ **Simple API for XML (SAX)** – събитийно ориентиран интерфейс, при който документът се чете последователно и неговото съдържание се рапортува като извиквания на отделни callback методи на обработващия обект (handler)
- ❖ **Document Object Model (DOM)** – зарежда и манипулира в паметта целия документ под формата на дърво
- ❖ Pull Parsing: **Streaming API for XML (StAX)** – третира документа като серия от възли, които се обхождат последователно с помощта на шаблона Iterator
- ❖ Свързване на данни (Data Binding): **Java Architecture for XML Binding (JAXB)** – дава възможност да се автоматизира процесът на запазване/извличане на данните от/в XML формат (**marshalling/unmarshalling**)
- ❖ Декларативни езици за трансформация като **XSLT** and **XQuery**.

DOM API: основни интерфейси (1)

[https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model]

- ❖ Attr
- ❖ CDATASection
- ❖ CharacterData
- ❖ Comment
- ❖ CustomEvent
- ❖ Document
- ❖ DocumentFragment
- ❖ DocumentType
- ❖ DOMException
- ❖ DOMImplementation
- ❖ DOMString
- ❖ DOMTimeStamp
- ❖ DOMStringList
- ❖ DOMTokenList
- ❖ Element
- ❖ Event
- ❖ EventTarget
- ❖ HTMLCollection

DOM API: основни интерфейси (2)

[https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model]

- ❖ MutationObserver
- ❖ MutationRecord
- ❖ NamedNodeMap
- ❖ Node
- ❖ NodeFilter
- ❖ NodeIterator
- ❖ NodeList
- ❖ NonDocumentTypeChildNode
- ❖ ParentNode
- ❖ ProcessingInstruction
- ❖ Selection*
- ❖ Range
- ❖ Text
- ❖ TextDecoder*
- ❖ TextEncoder*
- ❖ TimeRanges
- ❖ TreeWalker
- ❖ URL
- ❖ Window
- ❖ XMLDocument*

Благодаря за вниманието!



Trayan Iliev

**CEO of IPT – Intellectual Products
& Technologies**

<http://iproduct.org/>

<http://robolearn.org/>

<https://github.com/iproduct>

<https://twitter.com/trayaniliev>

<https://www.facebook.com/IPT.EACAD>

<https://plus.google.com/+IproductOrg>